

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

Appellant:	Ken Cameron) Confirmation No.: 5665
)
Serial No.:	10/074,019) Examiner: Asghar Bilgrami
)
Filed:	February 14, 2002) Art Unit: 2443
)
For:	METHOD FOR CONTROLLING THE ORDER OF DATAGRAMS) Date: March 21, 2010

APPEAL BRIEF

MAIL STOP APPEAL BRIEF - PATENTS

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This Appeal is from the decision of the Examiner dated April 21, 2009, repeating the rejection of pending claims 1-9 and 11-18, which are reproduced as the Claims Appendix of this brief.

I. REAL PARTY IN INTEREST

The assignee of record and real party of interest is Clearspeed Technology PLC.

II. RELATED APPEALS AND INTERFERENCES

There are presently no appeals or interferences known to the Appellant, the Appellant's representative, or the assignee, which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

III. STATUS OF CLAIMS

Claims 1-9 and 11-18 stand rejected and are the subject of this appeal. Claim 10 has been canceled.

IV. STATUS OF AMENDMENTS

Appellant filed a Request for Reconsideration on December 29, 2008. On April 21, 2009, the Examiner issued a second non-final Office Action, which maintained the pending rejection of claims 1-9 and 11-18.

V. SUMMARY OF THE CLAIMED SUBJECT MATTER

As described, starting on line 21 of page 3 of the application, the invention relates to a system and method for allocating datagrams or groups of datagrams to processing elements in that preserves global data packet order of the datagrams. The processing elements may be included in a multiple processing system (e.g., see page 4, lines 46-47). Datagrams are natural units of data in an application (e.g., see page 4, line 51 to page 5, line 1).

Datagrams are supplied from one or more data sources to a processing engine for processing by processing elements within the processing engine. Upon completion of the processing, the processing elements write the datagram to a data sink in the same order the datagrams were read from the data sources (e.g., see page 4, line 67 to page 5, line 13).

The forgoing features are recited in the claimed subject matter, which is described next with reference to the pending independent claims:

Independent Claim 1

Independent claim 1 recites a method for controlling the order of datagrams which are processed by at least one processing engine having at least one input port and at least one output port (e.g., see page 2, lines 35-39). The processing engine includes one or more processing elements (e.g., see FIG. 1, items 110a-110c and page 2, lines 46-47).

The method of claim 1 includes: each processor in the processing engine, once it becomes available, taking a ticket from a ticket dispenser, the ticket having a value associated therewith (e.g., see page 8, lines 45-46); waiting on an input buffer of the input port until the processor is given permission to continue according to the value of the ticket taken (e.g., see page 8, line 49); reading a next datagram or group of datagrams once the processor is given permission to continue (e.g., see page 8, line 50); signaling the input buffer for next ticket value (e.g., see page 8, line 51); processing the read datagram by the processor (e.g., see page 9, line 1); waiting on an output buffer of the output port until the processor is given permission to continue according to the value of the ticket taken (e.g., see page 9, line 2); writing the processed datagram once the processor is given permission to continue (e.g., see page 9, line 3); signaling the output buffer for next ticket value (e.g., see page 9, line 4); and repeating the previous steps for each ticket value (e.g., see page 9, line 5).

Independent Claim 5

Independent claim 5 is directed to a processing engine for processing datagrams in a predetermined order. The processing engine includes at least one input port, at least one output port (e.g., see page 2, lines 35-39) and a plurality of processing elements (e.g., see FIG. 1, items 110a-110c and page 2, lines 46-47).

Each processing element comprises an input port connected to the at least one input port of the processing engine (e.g., see page 3, lines 5-7), an output port connected to the at least one output port of the processing engine and arithmetic and logic means (e.g., see page 3, lines 8-9), and a ticket dispenser adapted to associate a ticket with each incoming datagram (e.g., see page 6, lines 20-23 and lines 40-41).

The processing elements, upon becoming available, take a next ticket from the ticket dispenser (e.g., see page 8, lines 45-46) an order of processing datagrams being controlled at the at least one input port of the processing engine and at the at least one output port of the processing engine in dependence on the ticket associated with the datagram or a group of datagrams (e.g., see page 8, lines 47-48) and the reading and writing of the read and processed datagram takes place upon the processor being given permission to continue (e.g., see page 8, line 49-50 and page 9, lines 2-3).

Independent Claim 8

Independent claim 8 is directed to a processing system that includes a plurality of processing engines for processing datagrams in a predetermined order. Each processing engine comprises at least one input port, at least one output port (e.g., see page 2, lines 35-39) and a plurality of processing elements (e.g., see FIG. 1, items 110a-110c and page 2, lines 46-47).

Each processing element comprises an input port connected to the at least one input port of the processing engine (e.g., see page 3, lines 5-7), an output port connected to the at least one output port of the processing engine and arithmetic and logic means (e.g., see page 3, lines 8-9), and a ticket dispenser adapted to associate a ticket with each incoming datagram (e.g., see page 6, lines 20-23 and lines 40-41).

The processing elements, upon becoming available, take a next ticket from the ticket dispenser (e.g., see page 8, lines 45-46) an order of processing datagrams being controlled at the at least one input port of the processing engine and at the at least one output port of the processing engine in dependence on the ticket associated with the datagram or a group of datagrams (e.g., see page 8, lines 47-48) and the reading and writing of the read and processed datagram takes place upon the processor being given permission to continue (e.g., see page 8, line 49-50 and page 9, lines 2-3).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

The grounds of rejection to be reviewed on appeal are as follows:

A. **Claims 1-9 and 11-18 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over U.S. Patent No. 6,338,078 to Chang et al. (hereinafter “Chang”).**

VII. ARGUMENT

A. **The rejection of claims 1-9 and 11-18 under 35 U.S.C. § 103(a), as allegedly being unpatentable over by Chang, should be reversed.**

Independent Claims 1, 5 and 8

The rejection based on the Chang document is believed to be erroneous as Chang does not teach, suggest or otherwise render obvious the combination of each and every feature recited in the claims. More specifically, the rejection is in error because Chang at least fails to disclose the features related to actions to be taken by the processor based on the value associated with a ticket taken from a ticket dispenser by the processor.

These actions, include, for example: (i) “waiting on an input buffer of the input port until the processor is given permission to continue according to the value of the ticket taken”; (ii) “reading a next datagram or group of datagrams once the processor is given permission to continue”; (iii) “waiting on an output buffer of the output port until the processor is given permission to continue according to the value of the ticket taken”; and (iv) “writing the processed datagram once the processor is given permission to continue” as recited in claim 1.

Similarly, Chang fails to disclose “writing of the read and processed datagram takes place upon the processor being given permission to continue” as recited in claims 5 and 8.

Chang discloses a computer network system that preserves packet order by using a hashing function to ensure maintenance of the packet sequence. Each packet is assigned to a queue and each queue is associated to a particular processor (e.g. see Chang, Fig. 3, queues 62-68 and CPUs 54-60 and col. 6, lines 20-36). Each queue corresponds to a particular processor and the particular processor only processes data packets from the corresponding queue.

The packets are assigned to the queues by a hashing function (col. 5, lines 14-17) based on the packet source MAC address, packet source IP address, or the packet' source and destination TCP port number, or all or a combination of these identifiers (col. 2, lines 64-67).

Applicant's independent claims 1, 5 and 8 are directed to a method, a processing engine and a system having a ticket dispenser that dispenses a ticket having a value associated therewith.

Claim 1 recites, *inter alia*, each processor taking a ticket from a ticket processor once the processor becomes available. Claims 5 and 8 recite, *inter alia*, a ticket dispenser associating a ticket with each incoming datagram and the processing elements taking a next ticket from the ticket dispenser upon (the processing element) becoming available.

Chang fails to teach or suggest processors, upon becoming available, taking the next ticket to process the associated datagrams. Chang does not even teach or suggest the use of a ticket dispenser which assigns a ticket to a datagram and a ticket being taken by a processor when that processor becomes available.

On page 8, lines 1 - 10 of the Office Action (of 21 April 2009), the Examiner alleges that it would have been obvious to use the hashing mechanism instead of a ticket dispenser.

The hashing mechanism of Chang places data packets in particular queues based on the source of the data packet. As described in Chang, if all data packets in a particular queue are processed, the processor corresponding to that particular queue will become dormant (col. 2, lines 62-64) even if data packets in other queues have not been processed (by processors corresponding to the other queues).

In contrast, as recited in claim 1, each processor takes a ticket from a ticket dispenser when the processor becomes available. That is, data packets may be fetched by any available processor. Consequentially, a processor will not remain dormant if data packets exist for processing.

Therefore, the hashing mechanism of Chang cannot be used as a ticket dispenser as alleged.

On page 8, lines 11 - 17 of this Office Action, the Examiner alleges that the “reading and writing of data packets” is an obvious function.

However, since Chang fails to disclose a ticket dispenser, Chang cannot be “waiting on an input buffer of the input port until the processor is given permission to continue according to the value of the ticket taken”. The hashing function of Chang, as described above, merely assigns a data packet to a particular queue based on the data packet source for processing by a particular CPU. There is no teaching or suggestion in Chang of a value being assigned to a ticket which provides the permission to continue.

Chang also fails to disclose “reading a next datagram or group of datagrams once the processor is given permission to continue” since the permission is based on the value of the ticket taken.

Similarly, Chang fails to disclose “waiting on an output buffer of the output port until the processor is given permission to continue according to the value of the ticket taken” and “writing the processed datagram once the processor is given permission to continue”

For at least the above reasons, Chang et al. document fails to teach, suggest or otherwise render obvious the combination of each and every feature recited in independent claim 1 and in independent claims 5 and 8. Accordingly, the § 103(a) rejection of claims 1, 5 and 8 is in error and should be reversed.

Dependent Claims 2-4 and 15-18

Claims 2-4 and 15-18 directly depend on independent claim 1, and thus at least include the combination of features recited in claim 1. Therefore, claims 2-4 and 15-18 are allowable at least for the reasons set forth above, and further for the additional recited features.

Dependent Claims 6 and 7

Claims 6 and 7 depend directly on independent claim 5, and thus at least include the combination of features recited in claim 5. Therefore, claims 6 and 7 are allowable at least for the reasons set forth above, and further for the additional recited features.

Dependent Claims 9 and 11-14

Claims 9 and 11-14 depend either directly or indirectly on independent claim 8, and thus at least include the combination of features recited in claim 8. Therefore, claims 9 and 11-14 are allowable at least for the reasons set forth above, and further for the additional recited features.

For at least the foregoing reasons, the Chang document does not teach, suggest or otherwise render obvious the combination of each and every feature recited in the claims. Accordingly, the pending rejections of the claims under 35 U.S.C. § 103(a) should be reversed.

Respectfully submitted,

Potomac Patent Group, PLLC

Date: 22 March 2010 /Krishna V. Kalidindi, Reg. No. 41,461/
Krishna V. Kalidindi (Reg. No. 41,461)

P.O. Box 270
Fredericksburg, Virginia 22404
(703) 893-8500
Fax: 919-361-0734
kris@ppglaw.com

VIII. CLAIMS APPENDIX

The following is a complete list of all claims on appeal:

1. A method for controlling the order of datagrams, the datagrams being processed by at least one processing engine, said at least one processing engine having at least one input port and at least one output port, the method comprises the steps of:
 - (a) each processor in the at least one processing engine, once it becomes available, taking a ticket from a ticket dispenser, the ticket having a value associated therewith;
 - (b) waiting on an input buffer of the input port until the processor is given permission to continue according to the value of the ticket taken in step (a);
 - (c) reading a next datagram or group of datagrams once the processor is given permission to continue;
 - (d) signaling the input buffer for next ticket value;
 - (e) processing the read datagram by the processor;
 - (f) waiting on an output buffer of the output port until the processor is given permission to continue according to the value of the ticket taken in step (a);
 - (g) writing the processed datagram once the processor is given permission to continue;
 - (h) signaling the output buffer for next ticket value; and
 - (i) repeating steps (a)-(h) for each ticket value.
2. A method according to claim 1, wherein the order of the datagram or group of datagrams at the at least one input port corresponds to the order of the datagrams at the at

least one output port.

3. A method according to claim 1, wherein the tickets comprise numerical values.

4. A method according to claim 1, wherein the ticket comprises a semaphore with data associated therewith.

5. A processing engine for processing datagrams in a predetermined order, the processing engine comprising:

at least one input port,

at least one output port and

a plurality of processing elements, each processing element comprising

an input port connected to the at least one input port of the processing engine,

an output port connected to the at least one output port of the processing

engine and arithmetic and logic means, and

a ticket dispenser adapted to associate a ticket with each incoming datagram,

wherein the processing elements, upon becoming available, take a next ticket from the

ticket dispenser an order of processing datagrams being controlled at the at least one

input port of the processing engine and at the at least one output port of the processing

engine in dependence on the ticket associated with the datagram or a group of

datagrams and the reading and writing of the read and processed datagram takes place

upon the processor being given permission to continue.

6. A processing engine according to claim 5, wherein the processing element comprises an element of a multi threaded array processing engine.

7. A processing engine according to claim 5, wherein the processing element can leave or enter the predetermined order.

8. A processing system comprising a plurality of processing engines for processing datagrams in a predetermined order, each processing engine comprising at least one input port, at least one output port and a plurality of processing elements, each processing element comprising an input port connected to the at least one input port of the processing engine, an output port connected to the at least one output port of the processing engine and arithmetic and logic means, and a ticket dispenser adapted to associate a ticket with each incoming datagram, wherein the processing elements, upon becoming available, take a next ticket from the ticket dispenser an order of processing datagrams being controlled at the at least one input port of the processing engine and at the at least one output port of the processing engine in dependence on the ticket associated with the datagram or a group of the datagrams and the reading and writing of the read and processed datagram takes place upon the processor being given permission to continue.

9. A processor system according to claim 8, wherein datagrams are processed in a round robin manner.

11. A processor system according to claim 9, wherein the tickets are issued on a first come first served basis.

12. A processor system according to claim 8, further comprising a counter for maintaining the value of the current ticket.

13. A processor system according to claim 12, wherein the counter comprises storage means for storing a numerical value.

14. A processor system according to claim 13, wherein once a processing element is allocated a datagram or group of datagrams for processing, the counter is incremented or decremented.

15. The method of claim 1, wherein a number of tickets is greater than a total number of the processors.

16. The method of claim 1, wherein the ticket represents an arrival time of the packet.

17. The method of claim 1, wherein the processor drops selected datagrams from being written to the output buffer.

18. The method of claim 1, wherein the processor enters or leaves a processing sequence.

IX. EVIDENCE APPENDIX

(None)

X. **RELATED PROCEEDINGS APPENDIX**

(None)